



Six Degree-of Freedom Haptic Rendering for Dental Implantology Simulation

Cedric Syllebranque, Christian Duriez

► To cite this version:

Cedric Syllebranque, Christian Duriez. Six Degree-of Freedom Haptic Rendering for Dental Implantology Simulation. International Symposium on Computational Models for Biomedical Simulation, Jan 2010, Phoenix, United States. pp.139–149, 10.1007/978-3-642-11615-5_15 . hal-00533097v2

HAL Id: hal-00533097

<https://hal.science/hal-00533097v2>

Submitted on 25 Feb 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Six Degree-of Freedom haptic rendering for dental implantology simulation

Cedric Syllebranque, Christian Duriez

¹ Didhaptic Laval, France

² INRIA Lille Nord Europe, University of Lille

Abstract. Training systems for dental implantology that are based on virtual reality need a precise haptic feedback with six Degrees of Freedom (6DoF). This is particularly challenging when using real patient data for the jawbone and during the interactive simulation of drilling. In this paper, we present our simulator and two complementary contributions: The first one is a method for 6DoF haptic rendering of contacts between the drilling tool and the jawbone model issued from a CT-scan. The second one is a fast simulation of the jawbone erosion during drilling which is relevant for 6DoF haptic rendering.

1 Introduction

During a dental implant procedure, the tooth root is replaced by an artificial one, in order to support restorations that resemble a tooth. The technique requires a planning, often based on a 3D CT scan, to identify the vital structures and to choose the appropriate orientation for the implants. The main difficulty during the surgery is the precise drilling of the jawbone.

A first pilot hole in the jawbone is expanded by using progressively wider drills, depending on chosen implant width and length. This must prevent from breaking the jawbone and avoids the elevation of the temperature under too strong drilling constraints, and thus a necrosis of the bone. The holes must be drilled on the accurate direction decided during the planning. However, the density of the bones is non homogeneous. The jawbone, like most of human bones, is composed of two different layers: the core, called spongy bone, is protected by a harder one, the cortical bone. Thus, when the cortical is about to be pierced (cortical breakthrough), the dentist must be careful that he does not apply an excessive force that would drive in the drill tool too deeply. One important skill is to be able to drill a precise hole even while being disturbed by the jawbone inhomogeneities. Moreover, as the view on the operative field is partially obstructed by the tools and cooling irrigation, the surgeons naturally use haptic feedback to perform the procedure.

The surgical placement of implants requires specific formation and the risk of complication decreases with the dentist experience [1]. The training usually begins by acquiring drilling experiences on some artificial jawbones made of plastic. Then, it quickly switches to companionship training on real patients.

Some expensive centers provide training on a cadaver head. The benefit of a training system based on virtual reality is to propose an alternative to training on cadaver. It would allow a repetitive rehearsal, and a wide choice of clinical cases.

To allow for acquiring the drilling skills described above, our simulator satisfies the following technical specifications. It includes a comprehensive virtual environment displaying important visual obstructions and anatomical obstacles due to the cheeks and the tongue. The data comes from real patients, so that the teacher can propose a large variety of pathologic cases. The interactions between the dentist tools and the jawbone, such as the collision response and the drilling, are simulated using a physics based approach. The system includes a haptic device with six Degree-of-Freedom (6DoF) (3 translation forces and 3 rotation torques) for rendering all forces that are applied to the drilling tool.

To satisfy these specifications, we have two contributions that are presented in this paper. We extend 6DoF god-object method proposed by [2] (see following section for details) to voxel based collision where the voxel map is issued from a CT-scan. Moreover, we propose a new method for the haptic rendering of a drilling task based on an efficient way of updating the CT-scan volume data.

The remainder of this paper is organized as follows. Section 2 summarizes previous work on 6 DoF haptic rendering and on drilling simulations. Section 3 presents the new algorithm for 6DoF haptic rendering of contacts based on jawbone models from CT-scan. The new method for the drilling simulation and haptic rendering is presented in section 4. Results are shown and discussed in section 5. Finally, we conclude in Section 6.

2 Previous Work

In this section, we only give a brief overview of previous work on haptic rendering of volumetric data including rendering of bone erosion and haptic rendering with 6 DoF. For more details, we refer a recent book on haptic rendering [3].

Voxel based haptic rendering with 6DoF was introduced in [4] with the Voxmap PointShell (VPS) software by Boeing. This pioneer work shows the efficiency of voxel-based collision detection. However, the proposed collision response, which is based on a computation of simple penalty forces, suffers from a lack of precision and stability. Several improvements (i.e. virtual coupling with pre-contact braking force) were proposed to avoid instabilities and improve haptic rendering (see also [5]) but the precision problem remains: the geometrical interpenetration at the end of the collision response depends on the stiffness of the contact springs which is limited by stability issues. In [6], authors propose to overcome the deficiencies of haptic hardware by augmenting perceptual rendering of forces associated to events. To improve precision and haptic rendering quality, [2] proposed a constraint-based approach which is an extension of the god-object method proposed in [7] to 6DoF haptic interaction. This algorithm is able to produce visually convincing non-penetrating configurations and thus improves the perceived stiffness of the objects [3]. However this method is based on

continuous collision detection algorithm between triangle meshes and is not applicable in conjunction with voxel-based collision detection. Several other works proposed the use of discrete distance fields in haptic rendering [8] [9] [10], but the rendering is based on penalty approaches which allows penetrating configurations. In the following section, we present an algorithm that extends 6DoF god-object method, based on constraints, to voxel-based collision method.

Haptic rendering of bone drilling has been studied in several papers. August *et al.* [11] developed an analytical model for bone erosion that can be parametrized with experimental data [12]. Their model is simple, fast to compute and fits quite well with experimental data. The geometry of the drill is assumed to be a sphere but Morris *et al.* [13] proposed a more complete erosion rate model based on sample points on the drill tool. An other milling model is also proposed in [14]. In [15] a discrete marching cube is performed in real-time to evaluate the contact forces. In [16] and [17] they propose a simulation of dental implantology based on FreeForm© software (from Sensable). However, in all these papers, haptic feedback is limited to 3DoF and is derived from elastic contact forces that are similar to penalty forces. In [18], drilling forces are derived from constraints and in [19] drilling forces are computed to ensure stable haptic feedback, but both methods are still limited to 3DoF rendering. In section 4, we propose a similar model of bone erosion but we couple it with a constraint-based approach for haptic feedback.

3 6DoF God-Object simulation on CT-scan model

The method described here extends the 6DoF constraint-based method proposed by Ortega [2] that was inspired by the pioneer work of Zilles and Salisbury [7]. The god-object method employs an idealized reference frame of the haptic device that is constrained to remain on the surface of the environment obstacles when the real haptic device can not prevent from penetrating the environment obstacles. At each time step, the algorithm computes the configuration that minimizes the distance between the real and the idealized reference frames, while meeting the constraints. The virtual object (here, the virtual drill) is attached to the idealized reference frame and the constraints are given by the collision between its geometry and the virtual environment geometry. This way, the user only sees realistic non-penetrating configurations between the virtual object and its environment. We provide an improvement to the method: our algorithm does not need continuous collision detection so that we can use voxel-based collision detection. Moreover, to include friction, we use another solving process.

3.1 Collision detection between jawbone and drill models

Our method relies on a 3D distance map of the jawbone. This map is precomputed off-line and updated on-line at high rates during drilling. It is extracted from a CT-scan of a real patient and discretized homogeneously using voxel

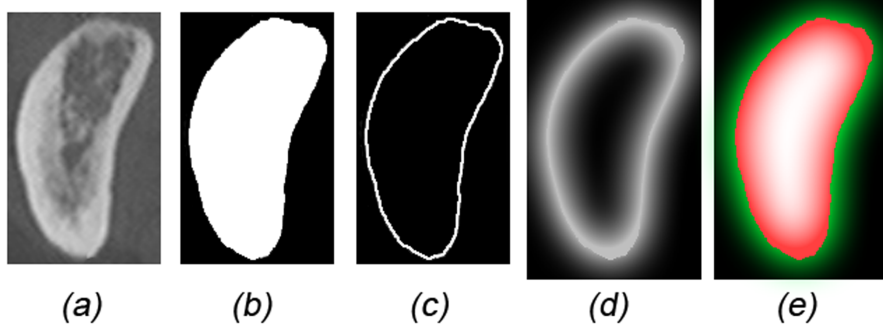


Fig. 1. *Illustration of 3D distance map extraction from CT-scan datas (here for one slice): (a) Original CT-scan slice - (b) Binarization - (c) Border extraction - (d) 3D distance map to surface - (e) Signed 3D distance map to surface (blue = outside, positive, and green = inside, negative)*

samples of a size about $0.2mm^3$. First, we binarize each CT-slice to obtain an inside/outside map. We assemble those binarized slices to get a volume block and then extract borders. Finally, for each of these border voxels, we compute a distance field in neighborhood (within a given distance d). Inside the volume, minimal distance are inverted. This process is illustrated by the figure 1. The result is a discrete potential field holding minimal distance to the object's surface that can be interpolated inside each voxel using a tri-linear interpolation. Using this approach, we obtain an implicit surface of the jawbone that can be evaluated very quickly (if χ is the potential function evaluated at point P , $\chi(P) < 0$ if and only if the point P is inside the jawbone).

For the collision detection with the virtual drill, we use the method proposed in [4]. The drill is modeled using pointshells: a set of surface point samples associated with inward-pointing surface normals. The collision computation consists in a simple evaluation of the potential function and the contact normal is taken from the pointshell normal. For each colliding point we associate a point on the jawbone surface, which is a *probable* entry point in the model of the jaw. The position of this surface point P_s is computed by raytracing technique between its colliding position P_c and its position P_{go} in the reference frame of the god object (with $\chi(P_{go}) \geq 0$).

As the collision technique is not continuous, we could miss some collision events. To avoid this problem we limit the possible displacement of the drill between two collision computations. We compute an in-between position for the drill tool, at the beginning of each time step, when the distance between the position of the god-object and the position of the interface exceeds a given threshold distance (in our case, the threshold equals the size of 6 voxels). The

³ CT-Scan resolutions vary from $0.1mm$ to $1mm$. If necessary, we resample the CT-Scan using trilinear interpolation.

colliding position P_c of each pointshell is computed in the in-between reference. Let \mathbf{X}_{go} and \mathbf{X}_r denote the 4×4 matrices describing, respectively, the position of the god-object frame and the real position of the interface. To obtain the in-between position \mathbf{X}_c , we interpolate the rigid displacement by using the same ratio for the interpolation of the translation and the rotation⁴.

3.2 Collision response based on friction contact

The way contacts are handled plays a very important role in the overall behavior of the interacting objects. The inclusion or not of a friction model highly influences the relative motion of the interacting objects and the haptic rendering. For each colliding pointshell, we compute a gap vector δ between the surface point P_s and the colliding position P_c . δ is projected along the normal \mathbf{n}_p and tangential directions \mathbf{t}_p and \mathbf{s}_p . Along the normal, Signorini's model is used and along the tangential directions, Coulomb's law is applied.

Our solving process is close to the one described in [20]. The mechanical coupling between pointshells comes from a mapping of a unique 6DoF damped spring⁵ (\mathbf{K}, \mathbf{B}) that links the god-object frame \mathbf{X}_{go} to the real device frame \mathbf{X}_r in the haptic control loop. \mathbf{H} provides the 3DoF motion dx of the pointshells due to a 6DoF motion $d\mathbf{X}$ of the god-object: $dx = \mathbf{H}d\mathbf{X}$. The compliance matrix of the pointshells \mathbf{W} is provided by: $\mathbf{W} = \mathbf{H}(\mathbf{K} + \mathbf{B}dt)^{-1}\mathbf{H}^T$

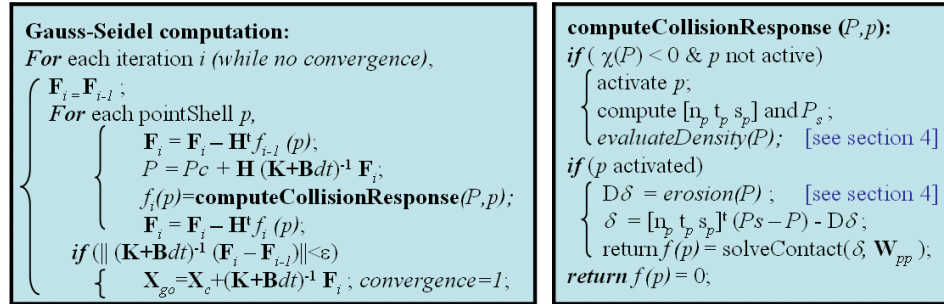


Fig. 2. Left: Gauss-Seidel algorithm. Right: collision response for each pointshell. \mathbf{W}_{pp} is the diagonal block of $\mathbf{H}(\mathbf{K} + \mathbf{B}dt)^{-1}\mathbf{H}^T$ for pointshell p .

We obtain a non-linear complementarity problem and the solving process is based on the iterative Gauss-Seidel algorithm. To optimize the computation, we do not compute the whole compliance matrix \mathbf{W} (that has a quadratic computational complexity): we only compute the diagonal blocks \mathbf{W}_{pp} and we use

⁴ Translation interpolation is linear. For interpolating the rotation, we use slerp.

⁵ The values in the 6×6 matrices \mathbf{K} and \mathbf{B} are tuned to optimize the haptic device mechanical stiffness and to ensure stability. The value of these matrices are given by the controller of the Virtuose haptic interface.

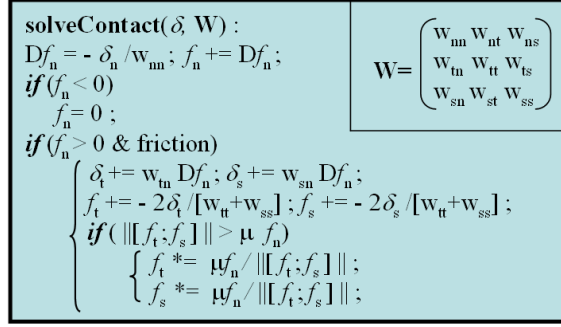


Fig. 3. *Computation of the contact force for one point-shell.*

operators \mathbf{H} and \mathbf{H}^T to evaluate, at each iteration and at each pointshell the influence other pointshells forces. The solving process is based on a Gauss-Seidel algorithm that is described in figures 2 and 3.

4 Drilling

In this section, we present a new method for haptic rendering of virtual drilling based on CT-scan data and compatible with the contact algorithm previously described. The outline of the method is illustrated in figure 4.

4.1 Erosion computation

Using a CT-Scan of a real patient jawbone and the method presented in [21], we compute a volumetric density map. The storage of the density map is optimized to shorten the time access using a sparse matrix indexed by 3 integers (i,j,k) to provide a direct memory access (a global transformation matrix which also includes scaling is used to position it in the virtual environment). We use this map to estimate the local density, i.e. the quantity of material that has to be removed to erode one voxel.

Like the contact algorithm, during the Gauss-Seidel iteration, we compute a collision detection by evaluating $\chi(P)$. We activate the pointshells that are colliding the jawbone and we compute their contribution to the drilling. The local erosion follows an analytical model that is close⁶ to the one proposed by [11]. After the integration of the erosion analytical model, we store a new density for each concerned voxel (in **evaluateDensity**(P) function in figure 2). If the density on a voxel vanishes, then we shrink the interpenetration distances by the size of this voxel (in **erosion**(P) function). Then we compute the Gauss-Seidel algorithm and the God-Object position the same way than for contact algorithm, except that the Coulomb's friction coefficient is canceled when drilling. As the

⁶ It only differs because the shape of our drill tool models are different.

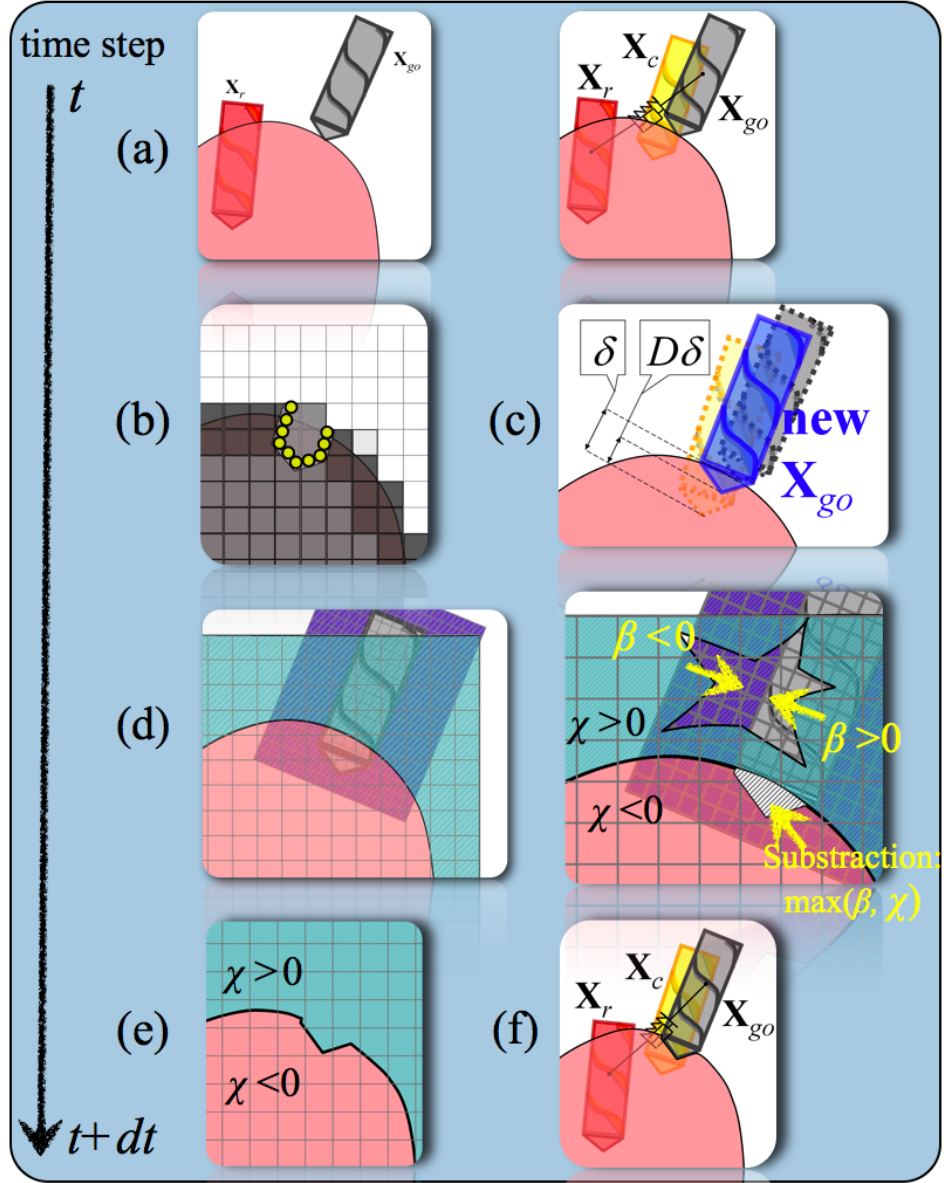


Fig. 4. Computation process for one time step: (a) In-between position \mathbf{x}_c is computed and collision is computed for the pointshells, - (b) For each colliding pointshell, an erosion rate is computed. - (c) The associated interpenetration distance is eroded. The computation continues with the Gauss-Seidel algorithm except that the drill tool can penetrate the jawbone inside with a depth equal to the shrinkage of δ . - (d) Afterwards, the distance map is updated using an algorithm close to the CSG subtraction. - (e) We obtain fast and precise geometrical results on the implicit surface. - (f) Finally, the algorithm restarts for a new time step.

interpenetration distances were eroded, the solution do not provide a collision-free configuration: the drill tool penetrates the jawbone surface according to the erosion computation.

During drilling, the drill tool rotates itself and its surface never sticks on the tooth surface. The friction is no more a stick-slip friction but a friction due to the erosion of the tooth surface by the drill tool. Coulomb's friction model is then replaced by a simple torque that is applied on the drill tool axis and is proportionnal to the quantity of eroded material.

4.2 On-line updates of the distance field

When the jawbone is eroded, we update the distance field map that we use for the collision computation and interpenetration evaluation. This process is performed as follows: we precompute another distance field map β for the the drilling tool⁷ (see figure 4), but we inverse the values of this map (positive values become negative and vice-versa). We then compute a bounding box around the drill map. For the voxels of the jawbone distance map that are inside this bounding box, we project their center in the tool inversed map. Let's $\beta(P)$ be the value found in the tool inversed map and let's $\chi(P)$ be the current value stored for this voxel. We take the maximum value and assign it to the voxel: $\chi(P) = \max(\chi(P), \beta(P))$. The obtained isosurface on the distance field map models precisely the erosion shape of the drill inside the jawbone. The algorithm can loop for a new time step.

5 Results

The presented algorithms are implemented on our virtual surgery training simulator solution. The haptic system is composed of an implantology hand-piece fixed on a *Haption 6DoF Desktop* device. The controller of the device is based on a virtual coupling scheme between the god-object frame and the real device frame. The virtual environment includes a volumetric jawbone, several different drills and a 3D patient in a 3D dental office (cf Fig. 5). We obtain precise and stable haptic feedback in 6DoF: to illustrate the precision, we focus on the force computation results and for stability, we show the computation time.

Haptic rendering. The figure 6 (b) shows the behaviour of our algorithm under a given user force (in red) applyied on a synthetic jawbone. This synthetic jawbone is made of a box of $1.1g/cm^3$ bone with a top cortical layer of $2.1g/cm^3$ (in (a) before drilling). The drilling depth (in blue) shows the contact/drilling algorithm resulting depth trough all the operation time. We can see that in the cortical part, the bone is hard to drill as it takes 6 seconds to pass through this $2.0mm$ layer, even when the user applies an increasing force reaching $15.0N$. However, after this layer's breakthrough, the drill goes faster and deeply in the

⁷ We use a method similar to the one we developed for the jawbone (cf Fig. 1), except that binarization step is made from the tool profile.

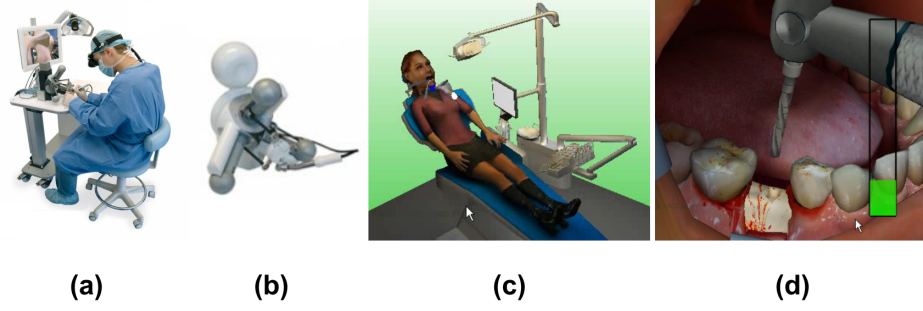


Fig. 5. Our surgery training simulator solution *VirTeaSy Implant Pro - Didhaptic*: (a) global view - (b) haptic device - (c) Virtual environment - (d) Virtual working zone closeup

soft spongy bone (5.0mm in 0.5s) while the feedback force felt by the user decreases quickly. This example shows the ability of our algorithms to reproduce the *cortical breakthrough* which must be avoided by the surgeon to prevent from cutting facial nerve passing into the bone.

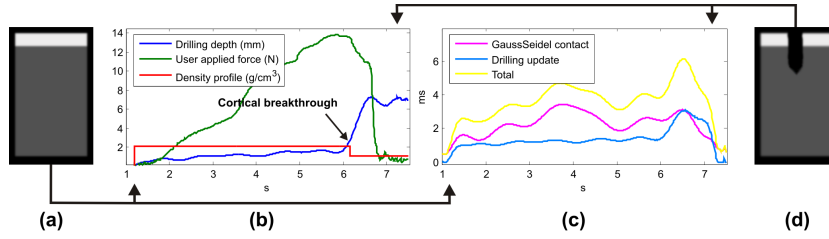


Fig. 6. Haptic rendering and computing times during drilling a synthetic jawbone with a 1.9mm drill. (a) a slice of the synthetic jawbone CT-scan before drilling. (b) drilling depth under a user force applied toward the bottom of the jawbone, with density profile. (c) computing times. (d) the same slice as (a) after the drilling.

Computation times. The computation times in the synthetic jawbone example are illustrated by the same figure (Fig. 6 (c)). The contact algorithm (in magenta) takes between 2ms and 3ms to converge on a 2.4 GHz Quad Core processor. At this maximum computation time, 63 pointshells were detected in collision. The rate obtained by the drilling part is related to the number of voxels involved in the distance map update. In the worst case, about 10000 voxels are in intersection with the tool and are updated in 3ms . Thus, the minimum total rate is near 166Hz for the simulation. To maintain a good stability, the control loop of the haptic device is still computed at 1000Hz . It uses the last position obtained for the god-object and the new position for the device and between

these two positions, the 6DoF damped spring is recomputed to obtain the new haptic force to display.

Precision. This contact algorithm rate is constrained by the precision needed for our application. Indeed, a drill must not be able to enter a hole made by a smaller one. The global convergence criterion must then be small enough to ensure such behaviour. In our tests, we achieve a precision of $10^{-2}mm$ with $0.2mm$ square voxels.

6 Conclusion and future work

This paper presents two technical contributions for haptic rendering in a training simulator of dental implantology. We presented an extension of the 6DoF god-object method to voxel-map collision and to contact with friction. We also proposed a new method for the 6DoF haptic feedback of drilling. Both methods are processed in a common manner and are totally compatible.

We plan to improve the performance by parallelizing the update of the distance map. In a more long term, we consider using a multi-threaded approach for the implementation of the Gauss-Seidel algorithm.

References

1. Binon, P.P.: Treatment planning complications and surgical miscues. *J. of Oral and Maxillofacial Surgery* **65**(7, Supplement 1) (2007) 73 – 92
2. Ortega, M., Redon, S., Coquillart, S.: A six degree-of-freedom god-object method for haptic display of rigid bodies. *Virtual Reality Conf., IEEE* **0** (2006) 191–198
3. Lin, M.C., Otaduy, M.A., eds.: *Haptic rendering: Foundations, algorithms and applications*. AK Peters (2008)
4. McNeely, W.A., Puterbaugh, K.D., Troy, J.j.: *Six degree-of-freedom haptic rendering using voxel sampling* (1999)
5. Petersik, A., Pflessner, B., Tiede, U., Hoehne, K.H., Leuwer, R.: *Haptic volume interaction with anatomic models at sub-voxel resolution* (2002)
6. Edmunds, T., Pai, D.K.: *Perceptual rendering for learning haptic skills*. In: *Proceedings of the Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Reno, Nevada, USA (2008)
7. Zilles, C.B., Salisbury, J.K.: *A constraint-based god-object method for haptic display* (1995)
8. Kim, L., Sukhatme, G., Desbrun, M.: *A haptic-rendering technique based on hybrid surface representation*. *Computer Graphics and Applications, IEEE* **24**(2) (March-April 2004) 66–75
9. W.A. McNeely, K.D. Puterbaugh, J.T.: *Voxel-based 6-dof haptic rendering improvements*. *Haptic-e* **3**(7) (january 2006)
10. Barlit, A., Harders, M.: *Gpu-based distance map calculation for vector field haptic rendering*. In: *WHC '07: Proceedings of the Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Washington, DC, USA, IEEE Computer Society (2007) 589–590
11. Agus, M., AndreaGiachetti, Gobbetti, E., Zannetti, G., Zorcolo, A.: *A multiprocessor decoupled system for the simulation of temporal bone surgery*. *Computing and Visualization in Science* (5) (2002) 35–43

12. Agus, M., Brelstaff, G.J., Giachetti, A., Gobbetti, E., Zanetti, G., Zorcolo, A., Picasso, B., Franceschini, S.S.: Physics-based burr haptic simulation: Tuning and evaluation. *Haptic Interfaces for Virtual Environment and Teleoperator Systems, International Symposium on* (2004) 128–135
13. Morris, D., Sewell, C., Barbagli, F., Salisbury, K., Blevins, N.H., Girod, S.: Visuo-haptic simulation of bone surgery for training and evaluation. *IEEE Computer Graphics and Application* (6) (2006) 48–57
14. Wang, S.M., Chiou, C.H., Cheng, Y.M.: An improved dynamic cutting force model for end-milling process. *journal mater. process. technol.* **148** (2004) 317–327
15. Marras, I., Papaleontiou, L., Nikolaidis, N., Lyroudia, K., Pitas, I.: Virtual dental patient: a system for virtual teeth drilling. In: *Multimedia and Expo, 2006 IEEE International Conference on*. (July 2006) 665–668
16. Kusumoto, N., Sohmura, T., Yamada, S., Wakabayashi, K., Nakamura, T., Yatani, H.: Application of virtual reality force feedback haptic device for oral implant surgery. *Clin. Oral Impl Res.* (17) (2006) 708–713
17. Ohtani, T., Kusumoto, N., Wakabayashi, K., Yamada, S., Nakamura, T., Kumazawa, Y., Yatani, H., Sohmura, T.: Application of haptic device to implant dentistry - accuracy verification of drilling into a pig bone. *Dental Materials journal* **28**(1) (2009) 75–81
18. Kim, L., Hwang, Y., Park, S., Ha, S.: Dental training system using multi-modal interface. *Computer-Aided Design and Applications* **2** (2005) 591598
19. Liu, G., Zhang, Y., Wang, D., Townsend, W.T.: Stable haptic interaction using a damping model to implement a realistic tooth-cutting simulation for dental training. *Virtual Reality* **12**(2) (2008) 99106
20. Duriez, C., dubois, F., Andriot, C., Kheddar, A.: Realistic haptic rendering of interacting deformable objects in virtual environments. *IEEE Transactions on Visualization and Computer Graphics* **12**(1) (2006) 36–47
21. Sjogreen, K., Ljungberg, M., Strand, S.E.: An Activity Quantification Method Based on Registration of CT and Whole-Body Scintillation Camera Images, with Application to 131I . *J Nucl Med* **43**(7) (2002) 972–982